

Fermilab Summer Software School
August 4 to 8, 2014
Overview of Supplementary Material
Draft Version

Scientific Software Infrastructure Department,
Fermilab Scientific Computing Division

August 3, 2014

Abstract

This handout provides an overview of supplementary material that has been prepared for the Summer 2014 edition of the Fermilab Software School. The supplementary material is intended for students whose computing skills are not yet at the level required for the Software School coursework. It introduces the student to C++ programming in the context of an event-processing framework called *art*. The *art* Workbook is a set of exercises designed around a toy experiment that allows self-paced training in the “art” of building physics programs within this framework. This handout includes excerpts from and references to it.

You can find this handout at:

<https://cdcv.s.fnal.gov/redmine/documents/783>

Contents

1	Introduction	2
2	Where to Start?	3
2.1	If you are new to HEP Software...	3
2.2	If you are an HEP Software expert...	3
2.3	If you are somewhere in between...	4
3	Links to Information	4
4	Guide to the Remainder of the Handout	4

1 Introduction

This course on programming reconstruction software for large computing projects assumes a fair bit of prior knowledge. For those of you that are just starting out in this area, some supplementary material is provided for you to work through.

The supplementary material is the *art* Workbook, a set of self-paced exercises that shows you how to use the *art* event processing framework, upon which several current experiments have built their physics software. The list of experiments using *art* include Darkside50, LBNE, MicroBooNE, mu2e, Muon g-2 and NOvA. If you belong to one of these, you will likely find the *art* Workbook particularly valuable.

To date, most high-energy physics experiments have written their code essentially from scratch and independently of each other. They certainly share some software analysis toolkits, but until recently there hasn't been a convenient, experiment-agnostic product to "plug into" that provides these bookkeeping tools.

By way of *art* (written in lower-case), the Scientific Computing Division provides a common, underlying software layer for experiments to build on, thus allowing you, the scientists, to concentrate on the physics code. *art* is a modular, C++-based event-processing framework that was adapted from the CMS framework, cmsrun.

An *art* Workbook is available in alpha release. It guides users through a set of exercises designed to illuminate the structure, user environment, configuration language and user-code requirements of *art*. It is intended to introduce much of the toolkit that you need to be an effective HEP programmer, including the C++ language, the C++ standard library, ROOT and CLHEP.

The Workbook currently includes introductory material and the first eight exercises of approximately three dozen planned. The authors hope and expect that experimenters will find the Workbook useful even at this early stage.

Please send your feedback — mistakes, confusing or missing material, navigation issues, and so on — to artists@fnal.gov. Constructive criticism will help us make it better!

2 Where to Start?

If you are ready to work through the course curriculum, do so. If not, we suggest that you begin working through the introductory portion of the *art* Workbook, some of which is excerpted in this document. Which portions of the *art* Workbook you can skip or skim depends on your current level of experience. We provide guidelines below.

2.1 If you are new to HEP Software...

Read Parts I and II (the introductory material and the Workbook) from start to finish. The Workbook is aimed at an audience who is familiar with (although not necessarily expert in) Unix, C++ and Fermilab's UPS product management system, and who understands the basic *art* framework concepts. The introductory chapters prepare the “just starting out” reader in all these areas.

2.2 If you are an HEP Software expert...

Read chapters 1, 2 and 3: this is where key terms and concepts used throughout the *art* document suite get defined. Skip the rest of the introductory material and jump straight into running Exercise 1 in Chapter 9 of the Workbook. Take the approach of: Don't need it? Don't read it.

2.3 If you are somewhere in between...

Read chapter 1,2 and 3 and skim the remaining introductory material in Part I to glean what you need. Along with the experts, you can take the approach of: Don't need it? Don't read it.

3 Links to Information

If you have an electronic version of this document, all cross-references and all external references (i.e. url's) in this section are live links.

This handout:

<https://cdcv.s.fnal.gov/redmine/documents/783>

Home page for the summer school:

<https://cdcv.s.fnal.gov/redmine/projects/fnal-soft-school-summer-2014/wiki>

Supplementary Materials:

https://cdcv.s.fnal.gov/redmine/projects/fnal-soft-school-summer-2014/wiki/Supplementary_Materials

Timetable for the summer school:

<https://indico.fnal.gov/conferenceDisplay.py?confId=8568>

Home page for the *art* Workbook:

<https://web.fnal.gov/project/artdoc/sitepages/home.aspx>

***art* Home page:**

<https://cdcv.s.fnal.gov/redmine/projects/art>

4 Guide to the Remainder of the Handout

The remainder of this handout contains the instructions for establishing your working environment and running the first couple of Workbook exercises. Most of it is excerpted from the Workbook; the setup information has been modified from the Workbook for this handout and can also be found at https://cdcv.s.fnal.gov/redmine/projects/fnal-soft-school-summer-2014/wiki/Supplementary_Materials.

The information is structured as follows:

- Supplementary Materials (page 7)
- Logging in and Establishing your Environment (page 7)
 - Using your Experiment’s Machines (page 7)
 - Using the *art* Workbook on Your OS X Mavericks laptop (page 8)
- From Section 9.6: Setting up to Run Exercise 1 (page 10)
 - Section 9.6.1: Log In and Set Up (page 10)
 - * Section 9.6.1.1: Initial Setup Procedure using Standard Directory (page 10)
 - * Section 9.6.1.3: Setup for Subsequent Exercise 1 Login Sessions (page 12)
- Section 9.7: Execute *art* and Examine Output (page 13)
- From Section 10.4: Initial Setup to Run Exercises (page 15)
 - Section 10.4.1: “Source Window” Setup (page 15)
 - Section 10.4.3 “Build Window” Setup (standard) (page 17)
- From Section 10.6: Running the Exercise (page 18)
 - Section 10.6.1 Run *art* on first.fcl (page 18)
- Chapter 11 General Setup for Login Sessions (page 19)
 - Section 11.1 Source Window (page 19)
 - Section 11.2 Build Window (page 20)

Note that the cross references in the excerpted pages don’t work; the section numbers are valid for Alpha Release v0_80 of the Workbook.

Several of the introductory sections are not reproduced here due to their length. They contain important background information that you will need in order to successfully complete the Workbook exercises. In particular,

- Chapter 3 *Introduction to the art Event Processing Framework* introduces you to many of the central concepts that *art* incorporates. Very important to read!

- Chapter 4 *Unix Prerequisites* is a must for those with little or no experience with Unix.
- Chapter 6 *Get your C++ up to Speed* outlines the features of C++ that you must master in order to use *art* effectively.
- The Workbook code, *art* and the software for your experiment all rely heavily on a variety of external software packages. Keeping dependencies straight and keeping up with changing versions requires the use of a product management system. Chapter 7 *Using External Products in UPS* provides information on the system used by *art*, called UPS.

Supplementary Materials

In addition to the primary course material, this course provides supplementary materials that can be used by students whose computing skills are not yet at the level required for the Software School coursework. It introduces the student to C++ programming in the context of an event-processing framework called art that is used by many of the experiments on which the students in this course work. The Workbook is a set of exercises designed around a toy experiment that allows self-paced training in the ``art'' of building physics programs within this framework.

On the first day of the course you will receive an handout that describes the supplementary material. The handout is also available online, at <https://cdcv.s.fnal.gov/redmine/documents/783> (Supplementary Material Handout; on that page, click on Supplementary.pdf).

The Workbook has a lot of material and not everyone needs to read every page. The handout provides guidance that suggest appropriate places to start, depending on what skills you already have.

Logging in and Establishing your Environment.

The workbook contains instructions on selecting machines and setting up your environment. That material is reproduced here for your convenience:

As for using the primary course materials, there are two options for using the supplementary materials:

- Your experiment's machines have been provisioned to build and run the course materials. You can log into one of those machines and work there.
- If you have a Mac laptop running Mavericks, you can download the required software, install it on your Mac and run the course exercises on your Mac.

Each is described below.

Using your Experiment's Machines

Darkside

```
ssh ds50.fnal.gov
source /ds50/app/ds50/ds50.sh
```

LBNE

```
ssh lbnegpvm06.fnal.gov
source /grid/fermiapp/lbne/software/setup_lbne.sh
```

MicroBoone and ArgoNeut

```
ssh uboonegpvm06.fnal.gov
source /grid/fermiapp/products/uboone/setup_uboone.sh # first option
# source /cvmfs/oasis.opensciencegrid.org/microboone/products/setup_uboone.sh # second option
```

Muon g-2

```
ssh gm2gpvm04.fnal.gov
source /grid/fermiapp/gm2/setup
```

Mu2e

```
ssh mu2egpvm05.fnal.gov
setup mu2e
```

NOvA

```
ssh novagpvm09.fnal.gov
# see information below for special instructions to set up environment
setup_art_workbook
setup_nova
# art v1_10_00b -q debug:e5:nu is now available
```

NOvA specific setup information

Paste the following function definition into your shell and execute it.

```
function setup_art_workbook
{
    echo "Setting up art workbook"
    source /grid/fermiapp/nova/novaart/novasvn/srt/srt.sh export EXTERNALS=/nusoft/app/externals
    export ART_WORKBOOK_OUTPUT_BASE=/nova/data/users export ART_WORKBOOK_WORKING_BASE=/nova/app/users
    export ART_WORKBOOK_QUAL="nu:e5"
```

```
}
```

If you do not have the `setup_nova` function defined, paste the following function definition into your shell and execute it.

```
function setup_nova {  
  echo ""  
  echo "NOVASoft"  
  echo ""  
  echo "Setting SRT_DIST, EXTERNALS"  
  echo "Sourcing generic setup_novasoft script for SVN control"  
  source /grid/fermiapp/nova/novaart/novasvn/srt/srt.sh  
  export EXTERNALS=/nusoft/app/externals  
  source $SRT_DIST/setup/setup_novasoft.sh "$@"  
  cd /nova/app/users/$LOGNAME  
  echo "Working directory: $PWD"  
}
```

Using the art Workbook on Your OS X Mavericks laptop

To run the art Workbook on your OS X Mavericks laptop you need to do two things:

- Install art as described on <https://cdcv.sfn.gov/redmine/projects/fnal-soft-school-summer-2014/wiki/FermiBasedCourseSetup>. The instructions are reproduced here for your convenience.
- Install the toyExperiment UPS product

Installing art

All the pre-built libraries and tools necessary for the course, including supplemental materials, can be easily installed on your OS X laptop. First obtain the pullProducts script from http://oink.sfn.gov/distro/art/pullProducts-v1_10_00b. You will execute this from a bash prompt on your laptop.

Please note this is still under rapid development. Please consult the online version of these instructions (at <https://cdcv.sfn.gov/redmine/projects/fnal-soft-school-summer-2014/wiki/FermiBasedCourseSetup>) for the most up-to-date information.

```
cd  
mkdir products  
cd products  
curl -O http://oink.sfn.gov/distro/art/pullProducts-v1_10_00b  
chmod +x pullProducts-v1_10_00b  
./pullProducts-v1_10_00b . dl3 artdev e5 prof # obtain optimized "profile" distribution
```

You now have the necessary products installed in `$HOME/products`.

To make use of them, you need to first make the tools available, and then to activate the appropriate products:

```
source $HOME/products/setups # can put this into the .bashrc or .bash_profile  
setup -B art v1_10_00b -q +e5:+prof
```

Detailed instructions are also available at http://oink.sfn.gov/distro/art/art-v1_10_00b.html.

Installing the toyExperiment UPS Product

This is still under development. Please consult the online version of these instructions (at https://cdcv.sfn.gov/redmine/projects/fnal-soft-school-summer-2014/wiki/Supplementary_Materials) for the most up-to-date information.

9.6 Setting up to Run Exercise 1

9.6.1 Log In and Set Up

The intent of this section is for the reader to start from “zero” and execute an *art* job, without necessarily understanding each step, just to get familiar with the process. A detailed discussion of what these steps do will follow in Section 9.9.

Some steps are written as statements, others as commands to issue at the prompt. Notice that *art* takes the argument `-c hello.fcl`; this points *art* to the run-time configuration file that will tell it what to do and where to find the “pieces” on which to operate.

Most readers: Follow the steps in Section 9.6.1.1, then proceed directly to Section 9.7.



If you wish to manage your working directory yourself, skip Section 9.6.1.1, follow the steps in Section 9.6.1.2, then proceed to Section 9.7.

If you log out and wish to log back in to continue this exercise, follow the procedure outlined in Section 11.

9.6.1.1 Initial Setup Procedure using Standard Directory

1. Log in to the computer you chose in Section 8.3.
2. Follow the site-specific setup procedure; see Table 5.1.
3. Make the standard working directory then `cd` to it; substitute your kerberos principal for the string *username*. These commands, shown on two lines, can each be typed on a single line.

```
mkdir -p $ART_WORKBOOK_WORKING_BASE/username/\  
workbook-tutorial/pre-built
```

```
cd $ART_WORKBOOK_WORKING_BASE/username/\  
workbook-tutorial/pre-built
```

4. Setup the toyExperiment UPS product:

```
setup toyExperiment v0_00_18 -q$ART_WORKBOOK_QUAL:prof
```

5. Copy the scripts into your working directory:

```
cp $TOYEXPERIMENT_DIR/HelloWorldScripts/* .
```

6. Use the provided script to create the symbolic links needed by the FHiCL files:

```
source makeLinks.sh
```

7. See what you have in the directory:

```
ls
bug01.fcl bug04.fcl inputFiles output
bug02.fcl helloExample.log inputs.txt
skipEvents.fcl bug03.fcl hello.fcl
makeLinks.sh writeFile.fcl
```

Proceed to Section 9.7.

9.6.1.3 Setup for Subsequent Exercise 1 Login Sessions

If you log out and later wish to log in again to work on this exercise, you need to do the following:

1. Log in to the computer you chose in Section 8.3.
2. Follow the site-specific setup procedure; see Section 5.
3. `cd` to your working directory, e.g., for the standard case (shown here on two lines):

```
cd $ART_WORKBOOK_WORKING_BASE/username/\  
workbook-tutorial/pre-built
```

4. Setup the toyExperiment UPS product:

```
setup toyExperiment v0_00_18 -q$ART_WORKBOOK_QUAL:prof
```

Compare this with the list given in Section 9.6.1. You will see that three steps are missing because they only need to be done the first time.

You are now ready to run *art* as you were before.

9.7 Execute *art* and Examine Output

From your working directory, execute *art* on the FHiCL file `hello.fcl` and send the output to `output/hello.log`:

```
art -c hello.fcl >& output/hello.log
```

Compare the output you produced (in the file `output/hello.log`) against Listing 9.1; the only differences should be the timestamps and some line breaking. *art* will have processed the first file listed in Table 9.1.

Listing 9.1: Sample output from running `hello.fcl`

```

1
2 %MSG-i MF_INIT_OK:  art 27-Apr-2013 21:22:13 CDT JobSetup
3 Messagelogger initialization complete.
4 %MSG
5 27-Apr-2013 21:22:14 CDT  Initiating request to open file
6 inputFiles/input01_data.root
7 27-Apr-2013 21:22:14 CDT  Successfully opened file
8 inputFiles/input01_data.root
9 Begin processing the 1st record. run: 1 subRun: 0 event: 1 at
10 27-Apr-2013 21:22:14 CDT
11 Hello World! This event has the id: run: 1 subRun: 0 event: 1
12 Begin processing the 2nd record. run: 1 subRun: 0 event: 2 at
13 27-Apr-2013 21:22:14 CDT
14 Hello World! This event has the id: run: 1 subRun: 0 event: 2
15 Hello World! This event has the id: run: 1 subRun: 0 event: 3
16 Hello World! This event has the id: run: 1 subRun: 0 event: 4
17 Hello World! This event has the id: run: 1 subRun: 0 event: 5
18 Hello World! This event has the id: run: 1 subRun: 0 event: 6
19 Hello World! This event has the id: run: 1 subRun: 0 event: 7
20 Hello World! This event has the id: run: 1 subRun: 0 event: 8
21 Hello World! This event has the id: run: 1 subRun: 0 event: 9
22 Hello World! This event has the id: run: 1 subRun: 0 event: 10
23 27-Apr-2013 21:22:14 CDT  Closed file inputFiles/input01_data.root
24
25 TrigReport ----- Event  Summary -----
26 TrigReport Events total = 10 passed = 10 failed = 0
27
28 TrigReport ----- Modules in End-Path: e1 -----
29 TrigReport  Trig Bit#   Visited    Passed    Failed    Error Name
30 TrigReport      0     0         10         10         0         0 hi
31
32 TimeReport ----- Time  Summary ---[sec]----
```

```
33 TimeReport CPU = 0.004000 Real = 0.002411
34
35 Art has completed and will exit with status 0.
```

Every time you run *art*, the first thing to check is the last line in your output or log file. It should be `Art has completed and will exit with status 0`. If the status is not 0, or if this line is missing, it is an error; please contact the *art* team as described in Section 3.4.

A future version of these instructions will specify how much disk space is needed, including space for all output files.

10.4 Initial Setup to Run Exercises

10.4.1 “Source Window” Setup

Up through step 4 of the procedure in this section, the results should look similar to those of Exercise 1. Note that the directory name chosen here in the `mkdir` step is different than that chosen in the first exercise; this is to avoid file name collisions.



If you want to use a self-managed working directory, in steps 3 and 4, make a directory of your choosing and `cd` to it rather than to the directory shown.

In your source window do the following:

1. Log in to the computer you chose in Section 8.3.
2. Follow the site-specific setup procedure; see Table 5.1.
3. Make a new working directory. Remember that you can type this command, and all subsequent commands in the Workbook that are shown on two lines for formatting reasons, on a single line.

```
mkdir -p $ART_WORKBOOK_WORKING_BASE/username/\
workbook
```

4. `cd` to the new directory.

```
cd $ART_WORKBOOK_WORKING_BASE/username/workbook
```

5. Set up the source code management system `git` and use it to pull down the workbook code to the directory `art-workbook`, which will be referred to as your *source* directory. The output for each step is explained in Section 10.4.2.1:

(a) `setup git`

(b) `git clone http://cdcv.s.fnal.gov/projects/art-workbook`

(c) `cd art-workbook`

(d) `git checkout -b work v0_00_24`

6. Source the script that sets up the environment properly:

```
source ups/setup_deps -p $ART_WORKBOOK_QUAL
```

The git commands are discussed in Section 10.4.2.1. The final step sources a script that defines a lot of environment variables — the same set that will be defined in the build window.

10.4.3 “Build Window” Setup

Again, advanced users wanting to manage their own working directory may skip to Section 10.4.3.2.



10.4.3.1 Standard Procedure

Now go to your build window and do the following:

1. Log in to the computer you chose in Section 8.3.
2. Follow the site-specific setup procedure; see Chapter 5.
3. `cd $ART_WORKBOOK_WORKING_BASE/username/\`
`workbook`
4. `mkdir build-prof`

The `build-prof` directory will be your *build* directory.

5. `cd build-prof`
6. `source ../art-workbook/ups/setup_for_development \`
`-p $ART_WORKBOOK_QUAL`

The space before the backslash is required here; there must be a space before the `-p`. The output from this command will tell you to take some additional steps; *do not do those steps*.

7. `buildtool`

This step may take a few minutes.

Skip Section 10.4.3.2 and move on to Section 10.4.4.

10.6 Running the Exercise

10.6.1 Run *art* on `first.fcl`

In your build window, make sure that your current working directory is your build directory. From here, run the first part of this exercise by typing the following:

```
art -c fcl/FirstModule/first.fcl >& output/first.log
```

(As a reminder, we suggest you get in the habit of routing your output to the `output` directory.) The output of this step will look much like that in Listing 9.1, but with two significant differences. The first difference is that the output from `first.fcl` contains an additional line

```
Hello from First::constructor.
```

The second difference is that the words printed out for each event are a little different; the printout from `first.fcl` looks like

```
Hello from First::analyze. Event id: run: 1 subRun: 0 event: 1  
while that from hello.fcl looked like
```

```
Hello World! This event has the id: run: 1 subRun: 0 event: 1
```

The reason for changing this printout is so that you can identify, from the printout, which module was run.

11 General Setup for Login Sessions

After you've done the initial setup described in Section 10.4, there are some steps that don't need to be repeated for subsequent login sessions. To begin a general login session for Exercise 2 or any subsequent exercise, you need to follow the instructions in this chapter.



If during your initial setup you chose to manage your own directory names, then the names of your source and build directories will be different than those shown here.

11.1 Source Window

In your source window:

1. Log in to the computer you chose in Section 8.3.
2. Follow the site-specific setup procedure; see Table 5.1
3. cd to your source directory:

```
cd $ART_WORKBOOK_WORKING_BASE/username/\  
workbook/art-workbook
```

4. Set up the environment:

```
source ups/setup_deps -p
```

The contents of the source directory is discussed in Section 10.4.2.2.

11.2 Build Window

In your build window:

1. Log in to the computer you chose in Section 8.3.
2. Follow the site-specific setup procedure; see Chapter 5
3. `cd` to your build directory:

```
cd $ART_WORKBOOK_WORKING_BASE/username/\  
workbook/build-prof
```

4. Source the setup file:

```
source ../art-workbook/ups/setup_for_development \  
-p $ART_WORKBOOK_QUAL
```

The build window setup is discussed in Section 10.4.4 and the *art* development environment is described in Section 10.5.